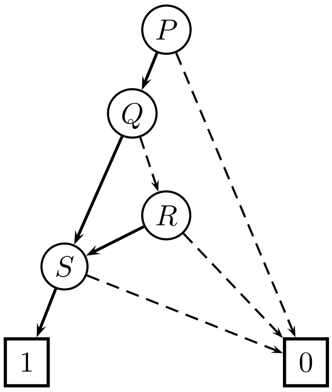
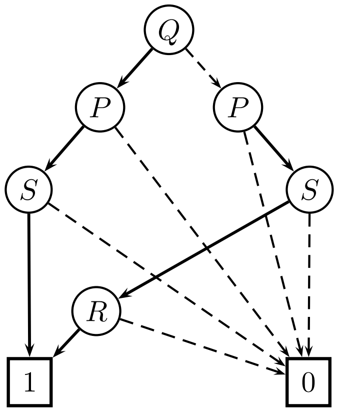
Automated Reasoning I, 2017
Final Exam, Sample Solution

Assignment 1

Part (a) With the ordering $P < Q < R < S$, we obtain the following OBDD for F:



Part (b) If the two variables that form the inner disjunction, namely Q and R, are the largest and the smallest element of the ordering, the OBDD needs two P-nodes and two S-nodes. E. g., for $Q < P < S < R$, we obtain



Part (c) By symmetry, we get a similar OBDD as in part (b) if we swap Q and R in the ordering and/or if we swap P and S in the ordering. This yields 4 different orderings with a 6-node OBDD.

Assignment 2

Part (a) Yes. Every inference of the modified calculus is also an inference of the unrestricted resolution calculus. Since the inference rules of the unrestricted resolution calculus are sound, the inference rules of the modified calculus are sound as well.

Part (b) No. The modified calculus is not refutationally complete. In particular, it is impossible to derive the empty clause from the clauses P and $\neg P \vee \neg P$, since the necessary resolution step violates the modified ordering restriction.

Assignment 3

The elements of the universe of $\mathcal{T}$ are the congruence classes of $\mathrm{T}_\Sigma(\{x, y, z\})$ with respect to E. There are five congruence classes, namely $\{x\}$, $\{y\}$, $\{z\}$, $\{c, d\}$, and a fifth class that contains all terms of $\mathrm{T}_\Sigma(\{x, y, z\})$ with f or b at the root. Consequently, we obtain:

(1) **true:** $[z] = \{z\}$.
(2) **false:** $[f(z)]$ contains b, $f(b)$, $f(f(b))$, ...
(3) **true:** $[c] = \{c, d\}$.
(4) **false:** $[f(c)]$ contains, e. g., $f(z)$.
(5) **true:** see above.
(6) **true:** $[b]$ is a congruence class.
(7) **false:** $\{x, y\}$ is not a congruence class.
(8) **false:** $z \approx f(x)$ is false for $\beta[z \mapsto [c]]$.

Grading scheme: 5th, 6th, 7th, 8th correct answer: 3 points each.

Assignment 4

Part (a) Since every LPO is a simplification ordering, we know that $s \succeq_{\text{lpo}} s'$. Therefore it suffices to show that $s' \succ_{\text{lpo}} t$. We prove this by induction over the structure of t:

If t is a variable, then by assumption $t \in \mathrm{var}(t) \subseteq \mathrm{var}(s')$; so $s' \succ_{\text{lpo}} t$ by case (1) of the LPO definition.

Otherwise $t = g(t_1, \dots, t_m)$ with $f \succ g$. By induction, $s' \succ_{\text{lpo}} t_j$ for all $1 \leq j \leq m$. So $s' \succ_{\text{lpo}} t$ by case (2)(b) of the LPO definition.

Part (b) Choose the precedence $f \succ g \succ h$, then $s = h(f(x), y) \not\succ_{\text{lpo}} g(y) = t$, but $f \succ g$ and $\mathrm{var}(s) \supseteq \mathrm{var}(t)$.

Assignment 5

Part (a) There are three critical pairs:

between (1) at position 1 and a renamed copy of (1):
  $\sigma = \{x \mapsto f(x')\}$,
  $g(b, f(x')) \leftarrow f(f(f(x'))) \rightarrow f(g(b, x'))$,
  critical pair: $\langle g(b, f(x')), f(g(b, x'))\rangle$.

between (2) at position 1 and (1):
  $\sigma = \{y \mapsto f(x)\}$,
  $f(h(f(x), y')) \leftarrow h(f(f(x)), y') \rightarrow h(g(b, x), y')$,
  critical pair: $\langle f(h(f(x), y')), h(g(b, x), y')\rangle$.

between (3) at position 1 and (2):
  $\sigma = \{y' \mapsto f(y),\ z \mapsto f(y)\}$,
  $h(f(y), b) \leftarrow g(h(f(y), f(y)), c) \rightarrow g(f(h(y, f(y))), c)$,
  critical pair: $\langle h(f(y), b), g(f(h(y, f(y))), c)\rangle$.

Grading scheme: 8 points; between -1 and -3 points per error.

Part (b) In order to avoid a critical pair between (1) and itself, we must orient (1) right-to-left.

In order to avoid a critical pair between (3) and (2), we must orient (2) right-to-left and (3) left-to-right.

There are several possibilities to orient the equations in this way using a KBO, for instance by choosing weight 1 for all function symbols and variables and a precedence $g > f > h > b > c$.

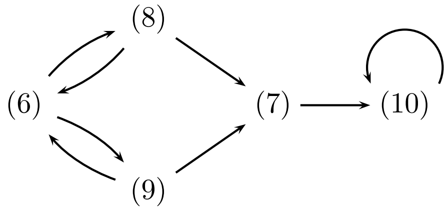
Grading scheme: 3 points for choosing an orientation without critical pairs; 3 points for defining a suitable KBO.

Assignment 6

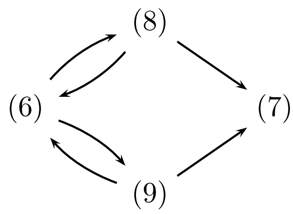
Part (a) The set of defined symbols is $D = \{f, g, k\}$, therefore R has five dependency pairs:

$$f^\sharp(x, y) \to g^\sharp(k(y), y) \qquad (6)$$
$$f^\sharp(x, y) \to k^\sharp(y) \qquad (7)$$
$$g^\sharp(h(x), y) \to f^\sharp(x, y) \qquad (8)$$
$$g^\sharp(x, h(y)) \to f^\sharp(x, y) \qquad (9)$$
$$k^\sharp(h(x)) \to k^\sharp(x) \qquad (10)$$

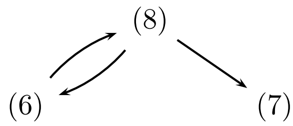
Part (b) The approximated dependency graph for R is



There are two SCCs with the node sets $\{(6), (8), (9)\}$ and $\{(10)\}$. We start with the second SCC and use a simple projection π with $\pi(k^\sharp) = 1$. For (10), $\pi(k^\sharp(h(x))) = h(x) \rhd x = \pi(k^\sharp(x))$. So we can delete node (10) from the dependency graph and obtain



For the remaining SCC $\{(6), (8), (9)\}$, we use a simple projection π with $\pi(f^\sharp) = 2$ and $\pi(g^\sharp) = 2$. For (6), $\pi(f^\sharp(x, y)) = y = \pi(g^\sharp(k(y), y))$, for (8), $\pi(g^\sharp(h(x), y)) = y = \pi(f^\sharp(x, y))$, and for (9), $\pi(g^\sharp(x, h(y))) = h(y) \rhd y = \pi(f^\sharp(x, y))$. So we can delete node (9) from the dependency graph and obtain



There is still one SCC left, namely $\{(6), (8)\}$. This SCC cannot be reduced using the subterm criterion. The SCC contains exactly one cycle $K = \{(6), (8)\}$.

Grading scheme: 3 points for the dependency graph; 3 points for the first node elimination; 2 points for the second node elimination.

Part (c) There are several K-minimal infinite rewrite sequences, for instance

$$\begin{tikzcd} f^\sharp(z, h(z)) \arrow[rr, "K"'] & & g^\sharp(k(h(z)), h(z)) \arrow[dl, "R"] \\ & g^\sharp(h(z), h(z)) \arrow[ul, "K"] & \end{tikzcd}$$

using (6), (4), and (8).